Behavioral Cloning: Convolutional Neural Networks to Transfer Learning from Humans to Machines

Pranit Gopaldas Shah[1], Hiral Pranit Shah[2]
[1] Dept. of Computer Science & Engineering,
Parul Institute of Engineering & Technology, Vadodara, India.
[2] Senior Data Analyst, TeerHub Technology Private Limited, Vadodara, India.

Abstract : Transferring learning from humans to machine has been a farfetched dream of human race. Human behavior is a confluence years of neurobiological learning. Spontaneous reaction to an event is embedded through firing of same neurons in the same sequence for countless times. Behavioral Cloning deals with the problem transferring the actions-reactions of a human to a machine. The machine is expected to imitate the actions-reactions of a human based on training. This paper investigates recent methods used in Behavioral cloning implemented specifically for autonomous driving. Methods were evaluated by comparing the techniques they rely on, type of work, use of theoretical proofs and simulations.

***IndexTerms* - Behavioral Cloning, Transfer Learning, Machine Learning, Computer Vision, Convolutional Neural Networks.**

I. Introduction

Human's ability to learn is fastest while copying, cloning or imitating someone or something. A policy is defined as a mapping of state to actions. Imitation Learning (IL) provides an appealing approach for autonomous driving: in many tasks, demonstrations of preferred behavior can be easily obtained from human experts, eliminating the need for expensive and potentially dangerous online data collection in the real world. Imitation Learning (IL) algorithms use expert data to train a parametric model which represents a policy. IL has two derived forms, Off-Policy architecture (Behavioral Cloning (BC)), and On- Policy architecture. States could be the sensor output in a vehicle and actions are the acceleration and steering angle. Off-Policy designs are data-driven and the illustrations are given independent of the policy of the robot, which could result in a high covariate shift error. This implies that the states encountered during testing are different from the ones encountered during training. On-Policy systems can help reduce the covariate shift as suitable feedback is provided by the human supervisor, but these methods can be unsafe and computationally expensive.

End-to-end Behavior cloning (Off-policy imitation learning) provides an alternative to traditional modular approach by simultaneously learning both perception and control using deep network. The network learns to recognize patterns associating sensory input (e.g., a single YUV image) with desired reaction in terms of vehicle control parameters producing a target maneuver. End-to-end behavior cloning eliminates the need for a fixed ontology and extensive amount of labeling. Finally, end-to-end imitative systems can be learned off-line in a safe way, in contrast to reinforcement learning approaches that typically require millions of trial and error runs in the target environment or a faithful simulation.

Here a qualitative analysis is made on the different CNN leaning models used for behavioral cloning for Autonomous Driving including their strengths and weaknesses. Further a proposal is made on the possible future trends of these systems.

The main problem statements in this paper can be outlined as follows:

- Identify Behavioral Cloning based Imitation technique for Autonomous Driving.
- Organize and analyze the approaches used for building behavioral cloning based Autonomous Driving systems.
- Define the strength and weakness of each system.
- Infer enhancements and enrichment that could be added to the systems.

Rest of the paper is organized as follows, Section I contains the introduction, main goal and problem statements, Section II talks about our motivation to undertake this study, Section III compares various methodologies used in recent research papers as a part of literature review, Section IV provides conclusion and future scope for the study undertaken.

II. Motivation

The main goal of this study is to explore some of the recent research in Behavioral-Cloning based Autonomous Driving or self-driving, to perform a feasibility study on recognizing human activity using hand movement analysis, to gather details of best practices in design and development of these innovative systems and to establish a base for further research. In this paper, we present a survey exploring the power and possibilities Vision base Hand Gesture Recognition in Human Computer Interaction techniques and also to study design issues and challenges in the area. Here a qualitative analysis is made on the different Hand Gesture Recognition systems to identify their strengths and weaknesses. Further a proposal is made on the possible future trends of these systems.

III. Literature Review

Here, we categorize seven of the recent research papers on Vision base Hand Gesture Recognition. In this section an overview of different architectural approaches used to build Hand gesture applications is given, with emphasis on research direction, technology and results from theoretical proofs or simulations.

One of the major downsides in the existing behavioral cloning based models is their prerequisite for an extensive training dataset which stems from the fact that such systems are designed to generalize the solution so that it works for a wide range of situations. NAVNet (Navigation Network) uses an off-policy imitation learning methodology for autonomous driving which is end-to-end trainable. It using a doubly-deep recurrent convolutional architecture that learns compositional representations in both space and time domains. The approach is non-data driven in nature and the system learns a regression-based mapping function between input images and steering angle. The model works by passing each visual input v_t through a feature transformation operation $\varphi V\ (v_t\ )$, which is parameterized by V to generate a fixed-length vector representation $\varphi_t \in R^d$. Once the feature-space representation of the visual input sequence is computed $(\varphi 1, \varphi 2, \varphi 3\ .\ .\ .\ , \varphi T)$, the sequential model takes over. The recurrent model, in a basic form, parameterized by W, transfers the input x_t and a previous hidden state h_{t-1} to an output z_t and a corresponding updates hidden state h_t , which implies that the inferences must be executed sequentially. The final step towards the prediction of a distribution $P(y_t\ )$ at a time step t, is to apply exponential linear unit (ELU) activation over the outputs z_t of the sequential model, thereby generating a possible distribution over a space C of possible per-time step outputs. Here, we take a late-fusion approach to merging the per time step predictions into a single prediction y for the input, which is essentially a temporal average of T predictions of one input frame. LRCN models are envisioned to become the upcoming standard methods for video description, activity recognition, autonomous navigation and any other tasks that require a deep understanding of both spatial and temporal characteristics [1].

Model-based reinforcement learning (MBRL) can plan to arbitrary goals using a predictive dynamics model learned from data, yet it estimates only what is possible and requires additional online data collection. A combination of imitation learning and model-based reinforcement learning (MBRL) learns preferred behavior by estimating the distribution of expert demonstrations, and then plans paths to achieve user-specified goals at test-time. This method provides a flexible, safe way to generalize to new goals by planning, compared to prior work on black-box, model-free conditional imitation learning. The algorithm produces an explicit plan within the distribution of preferred behavior accompanied with a score: the former offers interpretability, and the latter provides an estimate of the feasibility of the plan [2].

When human drivers navigate a vehicle through a road, they do not pay equal visual attention to everything that is present in their field of sight. As an example, let us consider, that there are two objects namely a traffic signal and a building present in the field of sight in front of a human driver, then the driver is most likely to pay more attention to the traffic signal rather than the building which is present out of the road. Visual attention to enhance autonomous driving performance is implemented in this novel method using unsupervised approach to train a model to learn to predict attention as it learns to drive a car. 3 models were generated. Model1: This model (henceforth referred to as Model1) was trained with original road scene images as input to predict the driving actions (steering angle, throttle and brake) as output. Model2: The input for this model (henceforth referred to as Model2) involved the incorporation of visual attention predicted by RoadSal. We multiplied the saliency value pixel-wise for each of the three channels in the original road scene (image). This saliency multiplied image was the input for Model2 and the output were the driving actions (steering angle, throttle and brake). Model3: This model (henceforth referred to as Model3) also used saliency multiplied images as in case of Model2 for input and the driving actions (steering angle, throttle and brake) were the output. The difference from Model2 was that in this case the predictions of Net1 (component of AutoTaskSal) were used as the saliency maps. Model1 has Mean Square Error of 0.01369, Model2 has Mean Square Error of 0.01145 and Model3 has Mean Square Error of 0.034. The fact that Model2 performs better than Model1 clearly demonstrates the usefulness of incorporating of task specific visual attention in the context of autonomous driving. The incorporation of visual attention indeed improved the performance of the autonomous driver. This can be attributed to original motivation that human drivers pay different levels of attention to various things in front of them while driving [3].

Current trend of the automotive industry combined with research by the major tech companies has proved that self-driving vehicles are the future. With successful demonstration of neural network based autonomous driving, NVIDIA has introduced a new paradigm for autonomous driving software. The biggest challenge for self-driving cars is autonomous lateral control. An end-to-end model seems very promising in providing a complete software stack for autonomous driving. Although this system is not ready to be provided as a feature in the market today, it is one of the many steps in the right direction to make self-driving cars a reality. The work described in this paper focusses on how an end-to-end model is implemented. The subtleties of training a successful end-to-end model are highlighted with the aim of providing an insight on deep learning and software required for neural network training. Detailed analyses of data acquisition and training systems are provided and installation procedures for all required tools and software discussed. TORCS is used for developing and testing the end-to-end model. Approximately ten hours of driving data was collected from two different tracks. Using four hours of data from a track, we trained a deep neural network to steer a car inside simulation. Even with such a small training set, the end-to-end model developed demonstrated capabilities to maintain lanes and complete laps in different tracks. For a multilane track, like the one used for training, the model demonstrated an autonomy of 96.62%. For single lane unknown tracks, the model steered the vehicle successfully for 89.02% of the time. With a small amount of training data, the network was able to successfully drive the car inside simulation [4].

Driving policies trained via imitation learning cannot be controlled at test time. A vehicle trained end-to-end to imitate an expert cannot be guided to take a specific turn at an upcoming intersection. This limits the utility of such systems. We propose to condition imitation learning on high-level command input. At test time, the learned driving policy functions as a chauffeur that handles sensorimotor coordination but continues to respond to navigational commands. We evaluate different architectures for conditional imitation learning in vision-based driving. We conduct experiments in realistic three-dimensional simulations of urban driving and on a 1/5 scale robotic truck that is trained to drive in a residential area. Both systems drive based on visual input yet remain responsive to high-level navigational commands. The controller that is trained using standard imitation learning only completes 20% of the episodes in Town 1 and 24% in Town 2, which is not surprising given its ignorance of the goal. More interestingly, the goal-conditional controller, which is provided with an accurate vector to the goal at every time step during both training and at test time, is performing only slightly better than the non-conditional controller, successfully completing 24% of the episodes in Town 1 and 30% in Town 2. Qualitatively, this controller eventually veers off the road attempting to shortcut to the goal. This also decreases the number of kilometers the controller is able to traverse without infractions. A simple feed-forward network does not automatically learn to convert a vector pointing to the goal into a sequence of turns. The proposed branched command-conditional controller performs significantly better than the baseline methods in both towns, successfully completing 88% of the episodes in Town 1 and 64% in Town 2. In terms of distance travelled without infractions, in Town 2 the method is on par with baselines, while in Town 1 it is outperformed by the nonconditional model. This difference is misleading: the nonconditional model drives more cleanly because it is not constrained to travel towards the goal and therefore typically takes a simpler route at each intersection. Applied the presented approach to vision-based driving of a physical robotic vehicle and in realistic simulations of dynamic urban environments. Results show that the command-conditional formulation significantly improves performance in both scenarios. While the presented results are encouraging, they also reveal that significant room for progress remains. In particular, more sophisticated and higher-capacity architectures along with larger datasets will be necessary to support autonomous urban driving on a large scale. We hope that the presented approach to making driving policies more controllable will prove useful in such deployment. Our work has not addressed human guidance of autonomous vehicles using natural language [5].

Controllable Imitative Reinforcement Learning (CIRL) successfully makes the driving agent achieve higher success rates based on only vision inputs in a high-fidelity car simulator. To alleviate the low exploration efficiency for large continuous action space that often prohibits the use of classical RL on challenging real tasks, our CIRL explores over a reasonably constrained action space guided by encoded experiences that imitate human demonstrations, building upon Deep Deterministic Policy Gradient (DDPG). Moreover, we propose to specialize adaptive policies and steering-angle reward designs for different control signals (i.e. follow, straight, turn right, turn left) based on the shared representations to improve the model capability in tackling with diverse cases. Extensive experiments on CARLA driving benchmark demonstrate that CIRL substantially outperforms all previous methods in terms of the percentage of successfully completed episodes on a variety of goal directed driving tasks. We also show its superior generalization capability in unseen environments. To our knowledge, this is the first successful case of the learned driving policy by reinforcement learning in the high-fidelity simulator, which performs better than supervised imitation learning. CIRL substantially outperforms all baseline methods under all conditions, especially better than their RL baseline. Furthermore, CIRL shows superior generalization capabilities in the rest three unseen setting (e.g. unseen new town), which obtains not perfect results but considerably better performance over other methods, e.g. 71% of our CIRL vs. 59% and 12% of IL and RL, respectively. More qualitative results provides some infraction examples that the IL model suffers from and CIRL successfully avoids. CIRL incorporates controllable imitation learning with DDPG policy learning to resolve the sample inefficiency issue that is well known in reinforcement learning research. Moreover, specialized steer-angle rewards are also designed to enhance the optimization of our policy networks based on controllable imitation learning. CIRL achieves the state-of-the-art driving performance on CARLA benchmark and surpasses the previous modular pipeline, imitation learning and reinforcement learning pipelines. It further demonstrates superior generalization capabilities on a variety of different environments and conditions [6].

Vision sensors like bio-motivated event-based cameras normally catch the elements of a scene, filtering out excess data. To make the best out of this sensor–calculation mix, cutting edge convolutional structures is adjusted to the yield of occasion sensors and broadly assess the presentation of our methodology on a freely accessible enormous scope occasion camera dataset (≈1000km). A lot of why a system delivers preferable outcomes on occasion pictures over on grayscale outlines is their capacity to catch scene elements. At high speeds, grayscale outlines experience the ill effects of movement obscure, while occasion pictures safeguard edge subtleties because of the high transient goals (microsecond) of occasion cameras and the way the positive and negative occasions in independent channels are procured that are taken care of to the system, accordingly keeping away from loss of data. The transient total expected to take care of the system does, notwithstanding, influence inactivity. Moreover, occasion cameras have an exceptionally high powerful range (HDR). Henceforth, occasion information speak to HDR substance of the scene, which is preposterous in conventional cameras since that would require long presentation times. This is beneficial so as to be powerful to various brightening conditions. Moreover, since occasion cameras react to moving edges and thusly filter out transiently repetitive information, they are more educational about the vehicle movement than individual grayscale outlines. DL-based methodology can benefit from the normal reaction of occasion cameras to movement and precisely foresee a vehicle controlling point under a wide scope of conditions [7].

Visual explanations take the form of real-time highlighted regions of an image that causally influence the network's output (steering control). Approach is two-stage. In the first stage, uses a visual attention model to train a convolution network end-to-end from images to steering angle. The attention model highlights image regions that potentially influence the network's output. Some of these are true influences, but some are spurious. Then apply a causal filtering step to determine which input regions actually influence the output. This produces more succinct visual explanations and more accurately exposes the network's behavior. This demonstrates the effectiveness of model on three datasets totaling 16 hours of driving. First showing that training with attention does not degrade the performance of the end-to-end network. Then showing that the network causally cues on a variety of features that are used by humans while driving. Model provides a better way to understand the rationale of the models decision by visualizing where and what the model sees to control a vehicle. A consecutive input raw images (with sampling period of 5 seconds) and corresponding attention maps $M_t = f_{map}(\{\alpha_{t,i}\})$. Three different penalty coefficients $\gamma \in \{0, 10, 20\}$, where the model is encouraged to pay attention to wider parts of the image with large γ. For better visualization, an attention map is overlaid by an input raw image and color-coded; for example, red parts represent where the model pays attention. For quantitative analysis, prediction performance in terms of mean absolute error (MAE) is explained on the bottom of each figure. The model is indeed able to pay attention on road elements, such as lane markings, guardrails, and vehicles ahead, which is essential for human to drive [8].

Brain-inspired cognitive model with attention (CMA) consists of three parts: a convolutional neural network for simulating human visual cortex, a cognitive map built to describe relationships between objects in complex traffic scene and a recurrent neural network that combines with the real- time updated cognitive map to implement attention mechanism and long-short term memory. The benefit of our model is that can accurately solve three tasks simultaneously: i) detection of the free space and boundaries of the current and adjacent lanes. ii) estimation of obstacle distance and vehicle attitude, and iii) learning of driving behavior and decision making from human driver. More significantly, the model accepts external navigating instructions during an end-to-end driving process. For evaluation, we build a large-scale road- vehicle dataset which contains more than forty thousand labeled road images captured by three cameras on our self-driving car. Moreover, human driving activities and vehicle states are recorded in the meanwhile. Cognitive model with attention, inspired by human brain, to simulate human visual and motor cortices for sensing, planning and control. The mechanism of attention modeled by a recurrent neural network in time. In addition, the concept of cognitive map for traffic scene was introduced and described in detail. Furthermore, a labeled dataset named Road-Vehicle Dataset (RVD) is built for training and evaluating. The performance of the model in planning and control was tested by three visual tasks. Experimental results showed that our model can fulfill some basic self-driving tasks with only cameras [9].

DARPA Autonomous Vehicle (DAVE-2) primary motivation is to avoid the need to recognize specific human-designated features, such as lane markings, guard rails, or other cars, and to avoid having to create a collection of "if, then, else" rules, based on observation of these features. The network consists of 9 layers, including a normalization layer, 5 convolutional layers and 3 fully connected layers. The input image is split into YUV planes and passed to the network. The first layer of the network performs image normalization. The normalizer is hard-coded and is not adjusted in the learning process. Performing normalization in the network allows the normalization scheme to be altered with the network architecture and to be accelerated via GPU processing. The convolutional layers were designed to perform feature extraction and were chosen empirically through a series of experiments that varied layer configurations. Uses strided convolutions in the first three convolutional layers with a 2x2 stride and a 5x5 kernel and a non-strided convolution with a 3x3 kernel size in the last two convolutional layers. Five convolutional layers are followed with three fully connected layers leading to an output control value which is the inverse turning radius. CNNs are able to learn the entire task of lane and road following without manual decomposition into road or lane marking detection, semantic abstraction, path planning, and control. A small amount of training data from less than a hundred hours of driving was sufficient to train the car to operate in diverse conditions, on highways, local and residential roads in sunny, cloudy, and rainy conditions. The CNN is able to learn meaningful road features from a very sparse training signal (steering alone). The system learns for example to detect the outline of a road without the need of explicit labels during training [10].

Table 1: Literature Review

Sr. No.	Paper Name	Year	Technique	Methodology	Advantage
1	Towards Behavioral Cloning for Autonomous Driving[1]	2019	1. PilotNet (CNN). 2. LSTM – Long Short Term Memory. 3. Long Recurrent Convolutional Network(LRCN) = CNN+LSTM	• RGB image capture at 30 frames/sec sized 160x320 • Trained for 100 epochs, learning rate 10^{-5}, MSE, Adam and ELU activation on NVIDIA GTX GeForce 1050 Ti. • Evaluation on simulator Udacity's Self Driving Simulator and KITTI dataset.	• Conventional method is improved by grasping both temporal and visual characteristics. • Architecture is easy to modify, train, test, needs minimal pre-processing with no hard-coded feature extraction.
2	Deep Imitative models for flexible inference, planning and control[2]	2019	1. Model-based Reinforcement Learning (MBLR) 2. Continuous-state, discrete-time, partially-observed Markov process Model. 3. CNN+RNN	• Training using CARLA on 25hr dataset of Town01. • Waypoint Planning, Cost palnning, Route Planning and Path Planning. • Featurized LIDAR to 200x200x2 • Noise Robustness and Reliability estimation. • Novel Obstacle avoidance.	• Interpretable expert-like plans without reward engineering. • Flexibility to new tasks • Robustness to goal specification noise: • Plan reliability estimation: • State-of-the-art CARLA performance
3	Visual Attention for Behavioral Cloning in Autonomous Driving[3]	2018	1. CNN to predict pixel-wise saliency/attention 2. RoadSal 3. AutoTaskSal	• Model1: Tobi Pro and OGAMA to record eye gaze data and Saliency Map Generation. • Model2:RoadSal:multiplied saliency value pixel-wise and pass through CNN to generate steering angle. • Model3:AutoTaskSal: RoadSal output as input and pass through CNN to generate steering angle.	• Incorporation of visual attention improves the performance of autonomous driving. • Depects humans pay differenct level of attention to various things in front of them while driving. • Performance improved by informing the driver about what is important for decision making through saliency map.
4	Behavioral Cloning for Lateral Motion Control of Autonomous Vehicles Using Deep Learning[4]	2018	1. Lambda Layer. 2. 2-D convolution using ReLU and stride of 5x5.	• Human driver steering using USB wheel. • TORCS 1.3.7 on Ubuntu 14.04 LTS and NVIDIA CUDA. • Street-1 and e-road in TORCS for data collection. • Image cropping, Bias removal, remap steering angle by factor of 5.	• Network was bale to complete full laps around different tracks indicating versality and validates the approach. • With a small amount of training data, the network was able to successfully drive the car inside simulation.

5	End-to-end driving via conditional imitation learning[5]	2018	1. Command Input Architecture= Image module(CNN) +measurement module(FCN) +command module (FCN). 2. Branched Architecture= Image module(CNN) +measurement module(FCN) +Branch(switch for each command).	• Data collection: 3 camera mounted on off-the-shelf 1/5 scale physical truck with an embedded NVIDIA TX2, flight controller (Holybro Pixhawk) running the APMRover firmware. • DataAugmentation: change in contrast, brightness, and tone, addition of Gaussian blur, Gaussian noise, salt-and-pepper noise, and region dropout, no geometric aug. • CARLA Unreal Engine 4. Town1 for training and Town2 for testing.	• Command-conditional formulation significantly improves performance • Generalization to new environments.
6	Controllable imitative reinforcement learning for vision based self-driving[6]	2018	1. Controllable Imitative Reinforcement Learning (CIRL)= imitation stage +Reinforcement stage(Actor-critic reward network) 2. Deep Deterministic Policy Gradient (DDPG) -an off-policy replay-memory -based actor-critic algorithm	• CARLA Unreal Engine 4. Town1 for training and Town2 for testing. • TensorFlow framework with training on four NVIDIA GeForce GTX1080 GPUs. • Learning and exploration rate linearly decreased to zero.	• First Successful deep-RL pipeline for vision-based autonomous driving • CIRL effectively alleviates the inefficient exploration of large-space continuous action space. • State-of-the-art performance on variety of scenarios and unseen environments.
7	Event-based vision meets deep learning on steering prediction for self-driving cars[7]	2018	1. 2D Histogram of positive and negative events 2. Event-to-frame conversion 3. Series of Resnet18 and ResNet50.	• Data produced by DVS sensor - asynchronous, pixel-wise brightness changes with very low latency and high dynamic range. • Event-to-frame conversion based on event polarity(-ve/+ve). • Synchronous event frame processed by ResNet-inspired network to produce steering.	• First large scale (1million images = 1000km)application of deep learning to event based vision on a regression task. • Leverage transfer learning from pre-tained convolutional network on classification tasks.
8	Interpretable Learning for Self-Driving Cars by Visualizing Causal Attention[8]	2017	1. Encoder: CNN feature extraction 2. Course-Grained Decoder: Visual Attention 3. Fine-Grained Decoder: Causality Test	• Data Source: 1200000 frames(=16hrs) from Comma.ai, Udacity and Hundai Centre of Excellence (HCE). • Effect of Combination of CNN+FCN/LSTM. • Effect of Penalty Coefficient, smoothing factor and causal visual saliency.	• Visual attention heat maps are suitable "explanations" for the behavior of a deep neural vehicle controller. • Attention maps comprise "blobs" that can be segmented and filtered to produce accurate maps of visual saliency.
9	Brain Inspired Cognitive Model with Attention for Self Driving[9]	2017	1. CNN Model for feature extraction using ReLu. 2. Segmentation-based approach. 3. RNN with LSTM blocks.	• Data: Diverse Visual data, Vehicle States data, Driver behavior data, Artificially tagged data. • Road-vehicle Dataset(RVD)	• Incorporate external controls through cognitive mapping model. • Some of the basic self-driving tasks are

				• Path planning and control through obstacles at 50,80,100,200,100 and 100 meter distances.	fulfilled with only cameras.
10	End to End Learning for self-Driving Cars[10]	2016	1. CNN Model. 2. Visualization of Internal CNN states.	• Data collection: camera placed on 2016 Lincoln MKZ or 2013 Ford Focus collecting 72 hours of data. • Data augmentation by adding artificial shifts and rotations. • 9 layer CNN including normalization, split into YUV. • Tested in Monmouth county, NJ.	• Learn the entire task of lane and road following without manual decomposition into road or lane marking detection, semantic abstraction, path planning and control. • Learns without the need of explicit labels during training.

IV. Conclusion And Future Scope

Autonomous Driving is the future of driving and from all of the researches we can envision that behavioral cloning (Imitation Learning) based convolutional models are the backbones. State-of-the-art results are achieved through a use of camera alone, however, some of the big challenges still remain open. Multi-agent dynamics and casual confusion are few of the challenges to be solved. Longitudinal control using vehicle speed along with steering angle is one more are to uncover. To improve the interaction between humans and machine, cars in this case, a Human guidance (voice guidance) using natural language seems to be a promising and still to be researcher.

References

[1] Kumaar, Saumya and Navaneethkrishnan, B and Hegde, Sinchana and Raja, Pragadeesh and Vishwanath, Ravi M (2019) Towards Behavioral Cloning for Autonomous Driving. In: IEEE INTERNATIONAL CONFERENCE ON ROBOTIC COMPUTING (IRC 2019), 25-27 Feb. 2019, Naples, Italy, pp. 560-567.

[2] Nicholas Rhinehart, Rowan McAllister, and Sergey Levine. Deep Imitative Models for Flexible Inference, Planning, and Control. pages 1-12, 2019. URL http://arxiv.org/ abs/1810.06544v4.

[3] Sourav Pal and Tharun Mohandoss and Pabitra Mitra. Visual Attention for Behavioral Cloning in Autonomous Driving. pages 1-12, 2018. URL http://arxiv.org/ abs/ 1812.01802v1

[4] S. Sharma, G. Tewolde and J. Kwon, Behavioral Cloning for Lateral Motion Control of Autonomous Vehicles Using Deep Learning, 2018 IEEE International Conference on Electro/Information Technology (EIT), Rochester, MI, 2018, pp. 0228-0233.

[5] F. Codevilla, M. Muller, A. Dosovitskiy, A. L opez, and V. Koltun. End-to-end driving via conditional imitation learning. arXiv:1710.02410, 2017.

[6] Liang X., Wang T., Yang L., Xing E. (2018) CIRL: Controllable Imitative Reinforcement Learning for Vision-Based Self-driving. In: Ferrari V., Hebert M., Sminchisescu C., Weiss Y. (eds) Computer Vision – ECCV 2018. ECCV 2018. Lecture Notes in Computer Science, vol 11211. Springer, Cham

[7] Maqueda, Ana I. et al. Event-Based Vision Meets Deep Learning on Steering Prediction for Self-Driving Cars. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2018): 5419-5427.

[8] Kim, Jinkyu & Canny, John. (2017). Interpretable Learning for Self-Driving Cars by Visualizing Causal Attention. 2961-2969. 10.1109/ICCV.2017.320.

[9] Chen, Shitao & Zhang, Songyi & Shang, Jinghao & Chen, BD & Zheng, Nanning. (2017). Brain Inspired Cognitive Model with Attention for Self-Driving Cars. IEEE Transactions on Cognitive and Developmental Systems. PP. 10.1109/TCDS.2017.2717451.

[10] Bojarski, Mariusz & Testa, Davide & Dworakowski, Daniel & Firner, Bernhard & Flepp, Beat & Goyal, Prasoon & Jackel, Larry & Monfort, Mathew & Muller, Urs & Zhang, Jiakai & Zhang, Xin & Zhao, Jake & Zieba, Karol. (2016). End to End Learning for Self-Driving Cars.

[11] Shah, Pranit & Pandya, Krishna & Shah, Harsh & Gandhi, Jay. (2019). Survey on Vision based Hand Gesture Recognition. International Journal of Computer Sciences and Engineering. 7. 281-288. 10.26438/ijcse/v7i5.281288.